Teslenko Artyom Romanovich
Student
Ural Federal University named after the first
President of Russia B.N. Yeltsin
Russia, Ekaterinburg
Academic supervisor: Ponomareva Elena Vladislavovna

NFC TECHNOLOGY AND ITS INFORMATION SECURITY

Abstract. *There is a separate variety in the theory of neural networks – these are neuro-fuzzy networks. They are based on the application of fuzzy logic and artificial neural networks in tandem. This device is recognized as the most promising for solving many problems of intelligent control. With it, you can both build regulators for systems and perform identification. In this article, we will examine various methods of training neuro-fuzzy networks by the example of training the most universal of them – ANFIS*

Keywords: *Neuro-fuzzy networks, ANFIS, adaptive neuro-fuzzy inference system, intelligent systems, membership function.*

Тесленко Артём Романович
Студент
Уральский Федеральный Университет имени первого
Президента России Б.Н. Ельцина
Россия, г. Екатеринбург
Научный руководитель: Пономарева Елена Владиславовна

МЕТОДЫ ОБУЧЕНИЯ НЕЙРО – НЕЧЕТКИХ СЕТЕЙ

Аннотация. *Существует отдельная разновидность в теории нейронных сетей – это нейро-нечеткие сети. Они основаны на применении нечеткой логики* *и искусственных нейронных сетей в тандеме. Этот аппарат признан самым перспективным для решения многих задач интеллектуального управления. С помощью него можно как строить регуляторы для систем, так и проводить идентификацию. В данной статье будут рассмотрены различные методы обучения нейро-нечетких сетей на примере обучения самой универсальной из них – ANFIS.*

Ключевые слова: *Нейро-нечеткие сети, ANFIS, адаптивно-сетевая система нечеткого вывода, интеллектуальные системы, функция принадлежности.*

INTRODUCTION

Once Stephen Hawking said: «The creation of artificial intelligence may be the last technological achievement of mankind if we do not learn to control risks.» These words describe the future of artificial intelligence, because artificial intelligence is developing very rapidly in our time. Many studies in the late 20th century focused on intellectual systems. Such systems include neuro-networks, fuzzy logic, evolutionary algorithms. Many of them in the 21st century were subject to hybridization, which allowed to strengthen advantages and offset disadvantages of individual systems.

The basis for the concept of artificial neural networks appeared when trying to study processes that occur in the brain during activity. The artificial neural network training algorithm was first proposed in 1949 by D. Hebb.

Fuzzy logic is one of the most powerful inventions of the 20century. Fuzzy logic uses state degrees in its set. Unlike classical logic, in which there are only two states of 1 and 0, in fuzzy logic the state falls into the interval [0; 1], that is, there can be both 0.15 and 0.73.

We will consider several approaches that allow us to successfully train a neuro-fuzzy network, in particular ANFIS. The approaches are universal and can be used to train the network to solve many problems.

Training of ANFIS

Backpropagation error algorithm

The first algorithm that we will consider will be called *backpropagation error algorithm*. It is one of many popular ANFIS training-algorithms.

The network requires prior training. The first phase of training is self-organization, in it the initial adjustment of parameters of membership functions and construction of rules base takes place. In the second phase the teacher is trained, in this phase the specified parameters are optimized.

The objective at this stage is:

Training data $x_i(t), i = 1,\dots,n$, desired outputs $y_i(t), i = 1,\dots,m$, fuzzy splits of variables $|T(x)|$ and $|T(y)|$, and a desired form of belonging function are given. We want to set the parameters of the membership function and find fuzzy rules.

At this stage, the network operates on two sides, that is, training input and output data are supplied to the network on two sides. The input and output spaces of each linguistic variable are divided into a given number of fuzzy sets, each of which has its own belonging function. Their parameters determine the shape and placement.

If you have training data and a fixed number of fuzzy sets, you can use the self-organizing learning method. The process of self-organizing training data automatically divides space into areas defining different groups of data. It determines the centers and width of the membership functions. The data grouped in the cluster is represented by a center point that defines the average value of all its elements. According to the algorithm, centers are located only in those areas of input and output spaces in which data is available. The number of clusters corresponds to the number of membership functions. You can use competitive learning to achieve these goals.

Before starting, the initial center values are randomly selected based on a uniform distribution. They have to be different.

For each data set we select the nearest center relative to the applied metric and adjust it according to the winner algorithm.

After you fix the locations of the centers, you set the parameters of the corresponding functions.

After determining the parameters of the functions, the values of the first and third layers can be calculated (taking into account that in the training mode the signal comes from both sides). The values of the second layer are calculated according to the relationships defined by the initial architecture. They determine the degrees of rule activity. Outputs of third layer determine degrees of reference signal belonging to fuzzy output sets.

Among all the links connecting the rule node (layer 2) to the term nodes of the output linguistic variables (layer 3), one with the maximum weight is selected, and the rest are excluded. Thus, the specific conditions of the rules are met by a single conclusion. If the weights of all links are very small, all links are removed, which means that the effect of this rule on the output variable is negligible, and the rule node is eliminated as a result.

Next, combine the rules, if possible, to reduce the number of rules. Once all the rules have been established, the network structure is finally defined.

The inverse error propagation method is used to teach with the teacher. The goal is to reduce the error function.

$$E = \frac{1}{2}\left(y(t) - \hat{y}(t)\right)^2,$$

where $y(t)$ is the desired output, $\hat{y}(t)$ is the current output.

In distinguishes from the usual method of inverse propagation of neural network error in a fuzzy neural network, the parameters of belonging functions are adjusted. For this purpose - we will implement stochastic gradient descent, that is, adjust them after each training example. To achieve a minimum of error, you need to "move" to the opposite side of the gradient.

Repeated training cycles lead to complete and fast network learning, especially if the initial values of the function parameters are close to optimal. Now the network is trained and ready to work with the task.

Annealing simulation algorithm

The second algorithm what we will consider called *annealing simulation algorithm.* An annealing simulation algorithm is a general algorithmic method for solving the problem of global optimization, especially discrete and combinatorial optimization.

The algorithm is based on the simulation of the physical process that occurs during the crystallization of the substance, including the annealing of metals. It is assumed that atoms have already lined up in a crystal lattice, but transitions of individual atoms from one cell to another are still permissible. It is assumed that the process proceeds at a gradually decreasing temperature. The transition of an atom from one cell to another occurs with some probability, and the probability decreases with a decrease in temperature. The stable crystal grid corresponds to the minimum energy of atoms, so the atom either moves to a state with less energy or remains in place.

$$P(X,X') = \begin{cases} 1 & F(X') - F(X) < 0 \\ exp\left(-\frac{F(X') - F(X)}{t_i}\right) & F(X') - F(X) \geq 0 \end{cases} .(1)$$

Algorithm for simulation of annealing for TSK network:

$$F(X) = \sum (y_j - y_j')^2 ,$$

where $\mathbf{y_j}$ is the output of the neural network, $\mathbf{y_j'}$ is the control values.

Conclusion

Thus, this article demonstrates two popular algorithms of training neuro-fuzzy networks on the example of training adaptive neuro-fuzzy inference system.

The algorithms are quite versatile and show themselves well in practice when solving various problems in different fields of science, from control in technical systems to application in medical robotics.

REFERENCES

1. Himansu Das, Bighnaraj Naik, H.S. Behera, Shalini Jaiswal, Priyanka Mahato, Minakhi Rout. Biomedical data analysis using neuro-fuzzy model with post-feature reduction. – 2020. – Text: electronic. – URL: https://www.sciencedirect.com/science/article/pii/S1319157819311656 (Reference date 16.09.2020).

2. Sasmita Acharya, C.R. Tripathy. An ANFIS estimator based data aggregation scheme for fault tolerant Wireless Sensor Networks. – 2016. – Text: electronic. – URL: https://www.sciencedirect.com/science/article/pii/S1319157816300775 (Reference date 30.09.2020)

3. Reza Kazemi, Majid Abdollahzade. Introducing an Evolving Local Neuro-Fuzzy Model – Application to modeling of car-following behavior. – 2015. – Text: electronic. – URL: https://www.sciencedirect.com/science/article/abs/pii/S0019057815002165 (Reference date 15.10.2020)

4. Chia-Hao Tu, Chunshien Li. Multitarget prediction using an aim-object-based asymmetric neuro-fuzzy system: A novel approach. – 2019. – Text: electronic. – URL: https://www.sciencedirect.com/science/article/abs/pii/S0925231220300473 (Reference date 1.11.2020)

5. Himansu Das, Bighnaraj Naik, H.S. Behera. Medical disease analysis using Neuro-Fuzzy with Feature Extraction Model for classification. – 2020. – Text: electronic. – URL: https://www.sciencedirect.com/science/article/pii/S2352914819302850 (Reference date 20.11.2020)